

CLAIMS

What is claimed is:

1. A method for class-based per-flow queuing across multiple hierarchical link-sharing structures, wherein each of said structures shares a single link, wherein each of said structures comprises a plurality of leaves, wherein each of said leaves represents a single flow, and wherein each of said leaves is common to each of said structures, the method comprising:
 - a) providing a leaf selection mechanism operative to enforce a plurality of rules adapted for said multiple hierarchical link-sharing structures; and
 - b) selecting in response to applying said leaf selection mechanism one of said leaves for servicing during a transmission opportunity.
2. A method according to claim 1 and further comprising updating operating parameters of said multiple hierarchical link-sharing structures to reflect resource usage by said leaf serviced during said transmission opportunity.
3. A method for class-based per-flow queuing across multiple hierarchical link-sharing structures, the method comprising:
 - a) distributing at least one of a plurality V of tokens to each of a plurality of hierarchical link-sharing structures, wherein each of said structures shares a single link, wherein each of said structures comprises a plurality of leaves, wherein each of said leaves represents a single flow, and wherein each of said leaves is common to each of said structures;
 - b) distributing each of said tokens to one of said leaves in each of said structures; and
 - c) selecting one of said leaves having at least V tokens for servicing during a transmission opportunity.
4. A method according to claim 3 wherein said distributing step a) comprises distributing a number of said tokens equal to the number of said structures.

5. A method according to claim 4 wherein said distributing step a) comprises distributing one of said tokens to each of said structures.

6. A method according to claim 3 wherein said distributing step a) comprises distributing a first number of said tokens to a first one of said structures having a first weighting and a second number of said tokens to a second one of said structures having a second weighting, wherein said numbers are relatively proportional to said weightings.

7. A method according to claim 3 wherein said distributing step b) comprises distributing to said leaf if said leaf has not exceeded its maximum allowable resource allocation for any of said structures.

8. A method according to claim 3 wherein said distributing step b) comprises distributing to said leaf if a blocking period is not currently in effect for said leaf.

9. A method according to claim 3 wherein said selecting step comprises selecting where a queue associated with said leaf has data ready to be serviced via said link.

10. A method according to claim 3 wherein said selecting step comprises selecting said leaf if said leaf has not exceeded its maximum allowable resource allocation for any of said structures.

11. A method according to claim 4 and further comprising debiting said serviced leaf by V tokens.

12. A method according to claim 3 and further comprising incrementing each of a plurality of usage counters at each node of said structures along the path from said serviced leaf to said link.

13. A method according to claim 3 wherein said selecting step comprises selecting said leaf from a group consisting of time-sensitive leaves.

14. A method according to claim 13 and further comprising:
time-stamping data upon arrival at each of a plurality of queues, where each of said queues is associated with one of said leaves; and
wherein said selecting step comprises selecting said leaf having the longest-waiting of said data where none of said time-sensitive leaves has a greater number of tokens than any other of said time-sensitive leaves.